# Part 2

In section 2, the modified versions of the djikstra and bellman ford algorithms are implemented, where each algorithm was allowed to perform at most k relaxations per node. Each implementation was then tested and compared against the original versions of the algorithms for accuracy, performance and memory usage. The results of the tests are shown in the tables below.

Accuracy

For the accuracy tests, the modified versions of the algorithms were tested against the original versions of the algorithms. Here are the results of the tests:

Figures 1 and 2 from the accuracy tests for the modified versions of the algorithms with $k = 1, ..., 10$. The results show that while the modified djikstra's algorithm has a 100% accuracy when compared to the original version of djikstra's algorithm, however, the lower the value of k, the less accurate the modified bellman ford algorithm is when compared to the original version of bellman ford.
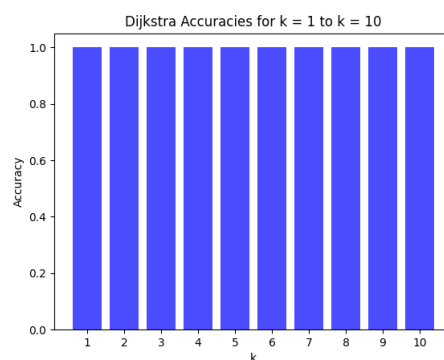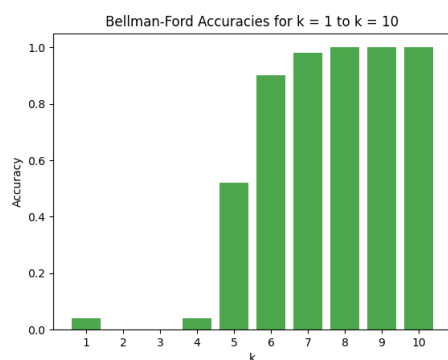
Time Complexity

Memory Usage

Part 3

In section 3, an algorithm to find the shortest path in a graph from all nodes to a single node were implemented. This was implemented in two different ways, one using djikstra's algorithm and the other using bellman ford.

For dense graphs, the single-source implementation of djikstra's algorithm has a time complexity of $O(V^2)$, where V is the number of vertices in the graph, while it's time complexity for sparse graphs is $O(E + V \log V)$, where E is the number of edges in the graph. The implementation of the shortest path algorithm (for all nodes) using djikstras has a time complexity of $O(V^3)$ for dense graphs and $O(E^2 + EV^2 \log V)$. This is because djikstra's algorithm is run V times, once for each node in the graph.

The same logic applies to the implementation of the shortest path algorithm using bellman ford. The time complexity for dense graphs is $O(V^3)$ and for sparse graphs is $O(E^2 + EV^2)$. The implementation of the shortest path algorithm using bellman ford has a time complexity of $O(V^4)$ for dense graphs and $O(E^2 + EV^2)$ for sparse graphs. This is because bellman ford is run V times, once for each node in the graph.